# Local Differentially Private Matrix Factorization with MoG for Recommendations

Jeyamohan Neera[1][0000−0001−8771−4193], Xiaomin Chen[1][0000−0001−9267−355X], Nauman Aslam[1], and Zhan Shu[2]

[1] Northumbria University, UK
jeyamohan.neera@northumbria.ac.uk
[2] University of Alberta, Canada

**Abstract.** Unethical data aggregation practices of many recommendation systems have raised privacy concerns among users. Local differential privacy (LDP) based recommendation systems address this problem by perturbing a user's original data locally in their device before sending it to the data aggregator (DA). The DA performs recommendations over perturbed data which causes substantial prediction error. To tackle privacy and utility issues with untrustworthy DA in recommendation systems, we propose a novel LDP matrix factorization (MF) with mixture of Gaussian (MoG). We use a Bounded Laplace mechanism (BLP) to perturb user's original ratings locally. BLP restricts the perturbed ratings to a predefined output domain, thus reducing the level of noise aggregated at DA. The MoG method estimates the noise added to the original ratings, which further improves the prediction accuracy without violating the principles of differential privacy (DP). With Movielens and Jester datasets, we demonstrate that our method offers a higher prediction accuracy under strong privacy protection compared to existing LDP recommendation methods.

**Keywords:** Local differential privacy · Matrix Factorization · Bounded Laplace Mechanism · Mixture of Gaussian

## 1 Introduction

Recommendation systems are often used to help users to find products or services that could interest them. Collaborative Filtering (CF) is a prominent technique used in recommendation systems. CF-based recommendation systems collect and analyse user information to offer better and personalized user experience. However, aggregation and analysis of user information can cause privacy violation. Narayanan et.al [13] demonstrated how analyzing an individual's historical ratings can reveal sensitive information such as user's political preference, medical conditions and even religious disposition. Therefore, it is crucial for recommendation systems to protect the privacy of the users while simultaneously providing high-quality recommendations.

Differential privacy (DP) has become a popular tool in various domains to protect the privacy of users even if the adversary possesses a substantial amount

of auxiliary information about the aggregated data [5]. Several studies have proposed differential privacy based CF mechanisms [11, 14, 18] to safeguard against privacy attacks in recommendation systems. However, most of the existing mechanisms imply that the data aggregator (DA) is trusted. Unfortunately, many DAs are inclined to collect more data than required and abuse the privacy of users for their benefits. Due to the concerns over untrusted DAs, many researchers [15, 16, 10, 1] have adopted Local Differential Privacy (LDP) for collaborative filtering. LDP based CF requires each user to locally perturb their data and sends the perturbed data to DA. However, this approach yields low prediction accuracy compared to DP based CF because each user's data is noised locally as opposed to adding noise to aggregates of the user's data. Therefore, it is necessary to design a LDP based recommendation system where each user can protect the privacy of their data from DA and at the same time, DA can perform recommendations with satisfactory prediction accuracy.

Our work aims to design a novel LDP based recommendation system which yields high data utility under strong privacy guarantee. We perturb user's original ratings locally using a Bounded Laplace mechanism (BLP) before sending to the DA. Furthermore, we reduce the prediction error by using MF with MoG at the DA. We estimate the added BLP noise using MoG [4], and Expectation-Maximization (EM) method is used to estimate the parameters of MoG. We demonstrate that our BLP-based recommendation system can provide substantial privacy protection and meanwhile achieve a satisfactory recommendation accuracy. The contribution of our work is as follows:

- We use a Bounded Laplace mechanism (BLP) to perturb each user's ratings locally in their devices. To the best of our knowledge, this is the first work which uses BLP to perturb each user's rating in recommendation systems. BLP ensures that the perturbed ratings fall within a predefined output domain without violating the principles of LDP. Additionally, BLP does not require complex computations to be performed in the user's side contrary to some existing solutions which require users to calculate their latent factors locally in their devices.
- We significantly improve the rating prediction accuracy of LDP based recommendation system. Local rating perturbation induces large error which grows linearly with the number of users and items. However, BLP compared to the Laplace mechanism introduces limited noise to aggregated ratings. Additionally, MoG is used to model the noise before MF to further increase the prediction accuracy. We demonstrate empirically using Movielens and Jester datasets that our proposed method can achieve satisfactory prediction accuracy under strong privacy guarantee and outperforms the works of [1] and [16].
- The communication cost of our proposed method is significantly less compared to other existing solutions, such as [16] as our method only requires users to transmit the perturbed ratings once to the DA, so there is no additional communication cost is introduced, unlike other methods that involve multiple iterations of information exchanges between a user and the DA.

## 2 Related Work

LDP is used to protect the user's privacy against untrusted DA in many applications. For example, Google uses LDP to collect each user's chrome usage statistics privately [6]. Likewise, LDP is also used in CF to protect the privacy of users. For instance, [15] introduced an LDP based rating perturbation algorithm which perturbs users' preference within an item category. Even though this mechanism hides a user's preference towards an item from an untrusted data aggregator, it can still reveal users' preferences towards an item category. Hua et.al [10] proposed another LDP based Matrix Factorization for untrusted DA. In their method initially, item profile vectors are learned using a private matrix factorization algorithm. Then these item vectors are sent to the user to derive user profile vectors. As each user's profile vectors do not depend on other users' data, they can easily compute their profile vectors locally instead of centrally. Users send their updated item profile vectors back to DA which then used to update the item profile vectors. The method used an objective function perturbation to achieve differential privacy. However, this method adds additional processing and communication overhead at user side.

Shin et.al [16] also proposed a method similar to [10] which requires the DA to send item profile vectors to each user. However, [16] used a randomized response perturbation mechanism instead of the objective perturbation mechanism and users send back the gradient instead of latent factors to DA. Their method also induces more communication and processing cost as users locally compute their user profile vectors over multiple iterations. Another LDP based rating perturbation mechanism was proposed by [1] where the original ratings are perturbed using Laplace mechanism. However, this proposed method used a clamping method to restrict the out-of-range ratings and used off-shelf optimization problems solvers such as SGD (Stochastic Gradient Descent) and ALS (Alternating Least Squares) in their MF algorithm.

## 3 Local Differential Privacy Based Recommendation System

In this work, we consider an untrustworthy data aggregator with whom the users are not willing to share any sensitive information. In our proposed system the original ratings are perturbed using Bounded Laplace mechanism and perturbed ratings are aggregated by DA. At DA, we use a MF with MoG for noise estimation and rating predictions. Our proposed rating prediction model will help the DA to reconstruct the original ratings from perturbed ratings without violating the privacy of users. Dwork et.al [5] proved that any mechanism that satisfies $\varepsilon$-differential privacy is resilient to post-processing. It implies that our perturbed rating from the local differentially private mechanism can be utilised in further processes without producing any additional privacy risk. Fig. 1 shows the system architecture for the proposed recommendation system.

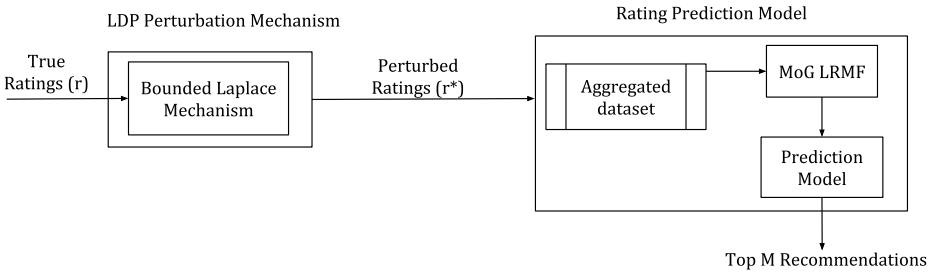



**Fig. 1.** Local differential privacy based recommendation

### 3.1 LDP Rating Perturbation

**Rating Normalization** As different recommendation systems use distinct rating scales, to produce a generalized theoretical model, we adopt the Min-Max scaling approach to normalize the rating scale between 0 and 1. Given an actual rating of $r^{\circ}$, the normalized true rating $r$ can be generated as:

$$r = \frac{r^{\circ} - r^{\circ}_{min}}{r^{\circ}_{max} - r^{\circ}_{min}} \tag{1}$$

in which $r^{\circ}_{max}$ is the highest possible rating score on the rating scale, and $r^{\circ}_{min}$ is the lowest. Local sensitivity is the maximum change rating perturbation mechanism can cause in a rating dataset, which is the difference between the maximum and the minimum rating. In a normalized dataset, the maximum rating is 1 and the minimum rating is 0. Therefore, the local sensitivity of the rating perturbation mechanism is $\Delta r = 1$.

### 3.2 Bounded Laplace mechanism

Our system perturbs the user's normalized rating using the BLP mechanism. Bounded Laplace mechanism is used to sanitize the output results of the Laplace mechanism with bounding constraints. BLP satisfies $\epsilon$-DP by ignoring out of bound values and re-samples noise for a given input rating $r$ until a value within the given bound is obtained. BLP mechanism can be defined as follows:

**Definition 1.** *(Bounded Laplace Mechanism) Given a scale parameter b and a domain rating interval of $(l, u)$, the Bounded Laplace mechanism $M_{BLP} : R \rightarrow R^*$ is given by a conditional probability density function as follows:*

$$f_{r^*|r}(r^*|r) = \begin{cases} \frac{1}{C_r(b)} \frac{1}{2b} exp(-\frac{|r^* - r|}{b}), & if\ r^* \in [l, u] \\ 0, & if\ r^* \notin [l, u] \end{cases} \tag{2}$$

*where $C_r(b) = \int_l^u \frac{1}{2b} exp(-\frac{|r^* - r|}{b}) dr^*$ is a normalization constant dependent on input rating $r$ and $r^*$ is the perturbed output.*

The normalization constant $C_r(b)$ can be given as:

$$C_r(b) = 1 - \frac{1}{2}\left(exp(-\frac{r-l}{b}) + exp(-\frac{u-r}{b})\right) \tag{3}$$

It can be easily proven that the result of integration will yield Eqn. (3). It has been shown in [9] that when local sensitivity $\Delta f = l-u$, BLP mechanism satisfies $\varepsilon$-local differential privacy. Using BLP in our proposed mechanism ensures that the perturbed output range is limited to $[l, u]$. However, the mechanism still guarantees that the adversary is unable to obtain any information about the original data by observing the output and thus preserves the privacy of the user. The privacy budget $\varepsilon$ will be determined by the DA and will be shared with user when they register with DA. The BLP mechanism (as given in Algorithm 1) will run every-time a user want to send rating to DA.

**Algorithm 1** Bounded Laplace Mechanism

**Input: Rating ($r$)** in
**Output: Perturbed Rating ($r^*$)** out
1: Generate a noise sample from the distribution $Lap(0, b)$
2: Calculate perturbed rating $r^* = r + Lap(0, b)$
3: **if** $l \leq r^* \leq u$ **then**
4: Set the perturbed rating to $r^*$
5: **else**
6: repeat Step 1
7: **end if**
8: **return** Perturbed rating to DA

### 3.3 Noise Estimation with MoG

Let $R_{m\times n}$ be the original normalized rating matrix and $R^*_{m\times n}$ be the perturbed rating matrix of $m$ users over $n$ items. The perturbed ratings can be decomposed as:

$$R^* = R + E \tag{4}$$

where $E_{m\times n}$ consists of BLP noise. Each element in the noised rating matrix can be represented as:

$$r^*_{ij} = r_{ij} + e_{ij} = (u_i^T)v_j + e_{ij} \tag{5}$$

where $u_i$ is a column vector in user latent factor matrix $U$ and $v_j$ is a column vector in item latent factor matrix $V$. As any unknown noise distribution can be modelled as a mixture of Gaussian, we assume that noise $e_{ij}$ in Eqn. (4) is drawn from MoG distribution [4]:

$$p(e_{ij} \mid \Pi, \Sigma) \sim \sum_{k=1}^{K} \pi_k \mathcal{N}(e_{ij} \mid 0, \sigma_k^2) \tag{6}$$

where $\Pi = (\pi_1, \pi_2, ....\pi_k)$, $\Sigma = (\sigma_1, \sigma_2, ....\sigma_K)$, $\sigma_k^2$ is the variance of Gaussian component $k$ and $K$ is the total number of Gaussian components. $\pi_k$ is the mixing proportion and $\sum_{k=1}^{K} \pi_k = 1$. Therefore, the probability of each perturbed rating $r_{ij}^*$ of $R$ can be represented as:

$$p(r_{ij}^* \mid u_i, v_j, \Pi, \Sigma) = \sum_{k=1}^{K} \pi_k \mathcal{N}(r_{ij}^* \mid (u_i^T)v_j, \sigma_k^2) \tag{7}$$

The likelihood of $R^*$ can thus be given as:

$$p(R^* \mid U, V, \Pi, \Sigma) = \prod_{i,j \in \Omega} \sum_{k=1}^{K} \pi_k \mathcal{N}(r_{ij}^* \mid (u_i^T)v_j, \sigma_k^2) \tag{8}$$

where $\Omega$ is the set of non-missing data points in perturbed rating matrix $R^*$. Given a dataset $R^*$, our goal is to compute the parameters $U, V, \Pi$ and $\Sigma$ such that the maximum log-likelihood of $R^*$ is achieved.

$$\begin{aligned} &\max_{U,V,\Pi,\Sigma} \log p(R^* \mid U, V, \Pi, \Sigma) \\ &= \sum_{i,j \in \Omega,} \log \sum_{k=1}^{K} \pi_k \mathcal{N}(r_{ij}^* \mid (u_i^T)v_j, \sigma_k^2) \end{aligned} \tag{9}$$

### 3.4 Expectation Maximization for MoG

As maximum log-likelihood function given in Eqn.(9) cannot be solved using a closed-form solution, Expectation-Maximization (EM) algorithm is used to estimate model parameters $U, V, \Pi$ and $\Sigma$. The EM algorithm introduced in [4] has two steps, Expectation and Maximization. In E-step we compute posterior responsibility using the current model parameters $U, V, \Pi$ and $\Sigma$ for each noise point $e_{ij}$ as:

$$\gamma_{ijk} = \frac{\pi_k \mathcal{N}(r_{ij}^* \mid (u_i^T)v_j, \sigma_k^2)}{\sum_{k=1}^{K} \pi_k \mathcal{N}(r_{ij}^* \mid (u_i^T)v_j, \sigma_k^2)} \tag{10}$$

The posterior responsibility reflects the probability that it is Gaussian component $k$ generates the noise data point $e_{ij}$. In M-step we re-estimate each model parameter $U, V, \Pi, \Sigma$ using the posterior responsibilities such that the maximum log-likelihood Eqn.(11) is obtained [12].

$$\max_{U,V,\Pi,\Sigma} \sum_{i,j \in \Omega} \sum_{k=1}^{K} \gamma_{ijk} \left( \log \pi_k - \log \sqrt{2\pi}\sigma_k - \frac{(r_{ij}^* - (u_i^T)v_j)^2}{2\sigma_k^2} \right) \tag{11}$$

To solve the problem given in Eqn.(11), we first update $\Pi$ and $\Sigma$:

$$N_k = \sum_{\forall i,j} \gamma_{ijk}$$

$$\pi_k = \frac{N_k}{N}$$

$$\sigma_k^2 = \frac{1}{N_k} \sum_{\forall i,j} \gamma_{ijk} (r_{ij}^* - (u_i^T) v_j)^2 \tag{12}$$

where $N_k$ is the sum of posterior responsibilities for $k$th Gaussian component and $N$ is the total number of data points. The portion of Eqn. (11) related to $U$ and $V$ can be rewritten as:

$$\begin{aligned} &\sum_{i,j \in \Omega} \sum_{k=1}^{K} \gamma_{ijk} \left( - \frac{(r_{ij}^* - (u_i^T) v_j)^2}{2\sigma_k^2} \right) \\ =& - \sum_{i,j \in \Omega} \left( \sum_{k=1}^{K} \frac{\gamma_{ijk}}{2\sigma_k^2} \right) ((r_{ij}^* - (u_i^T) v_j)^2) \\ =& - || \, W \odot (X - UV^T) \, ||_{L_2} \end{aligned} \tag{13}$$

where $W$ is the weight matrix in which the element $w_{ij}$ is the weight for rating $r_{ij}$ and can be defined as:

$$w_{ij} = \begin{cases} \sqrt{\sum_{k=1}^{K} \frac{\gamma_{ijk}}{2\sigma_k^2}}, & \text{if } i,j \in \Omega \\ 0, & \text{if } i,j \notin \Omega \end{cases} \tag{14}$$

The problem defined by Eqn.(13) is equivalent to a weighted L2 low rank matrix factorization problem and any weighted L2-norm solvers such as WPCA [3], WLRA [17] and DN [2] can be used to solve it. We used WPCA in our evaluation. The process of our noise estimation and rating prediction is given in Algorithm 2. The convergence is achieved when the change between two consecutive $U$ latent factor matrices is smaller than a predefined threshold or if the maximum number of iterations is reached.

**Algorithm 2** MoG based Noise Estimation and Prediction

**Input: Noised Ratings ($R^*$)** in
**Output: Predicted Rating** out
*Initialization* : random initialization of $U, V, \Pi$ and $\Sigma$
1: (E-Step) Estimate posterior responsibility $\gamma_{ijk}$ using Eqn.(10)
2: **for** until convergence occurs **do**
3: (M-Step for $\Pi$ and $\Sigma$) Estimate MoG parameters $\Pi$ and $\Sigma$ using Eqn.(12)
4: (M-Step for $U, V$) Estimate $U$ and $V$ by solving Eqn.(13)
5: **end for**
6: **return** Predict ratings using inner product of user and item latent factor matrices $U$ and $V$

## 4 Evaluation

In this section, we discuss the evaluation of our proposed BLP based MF with MoG approach (BLP-MoG-MF). To demonstrate the effectiveness of our proposed approach, we compare it with the following methods:

- Non-Private Matrix Factorization (Non-Private MF): This is the baseline method we compare our approach with. This method does not perturb any user's ratings and uses SGD based matrix factorization for rating prediction. RMSE value of the baseline method reflects the lower bound for prediction error that can be obtained without any privacy constraints.
- Input Perturbation SGD Method (ISGD) [1]: ISGD method perturb ratings using Laplace mechanism and clamp the resulting perturbed ratings using a clamping parameter locally at the user's device. DA uses MF with SGD method for rating prediction.
- Private Gradient-Matrix Factorization (PG-MF) [16]: In this method initially DA computes item latent factors and sends to each user. Then each user computes their latent factors locally in their device and submits a perturbed gradient to DA. DA updates the item latent factors using the aggregated perturbed gradients from each user.

### 4.1 Datasets

We used two popular public rating datasets in our evaluation: Movielens [8] and Jester [7]. Among several different version of Movielens dataset, we used the dataset which consists of 100k ratings of 1682 movies rated by 943 users. The minimum rating given is 0.5 and the maximum rating is 5. The Jester dataset consists of 2M ratings of 100 jokes rated by 73,421 users. The minimum rating given in this dataset is -10 and maximum rating given is +10.

### 4.2 Evaluation Metrics

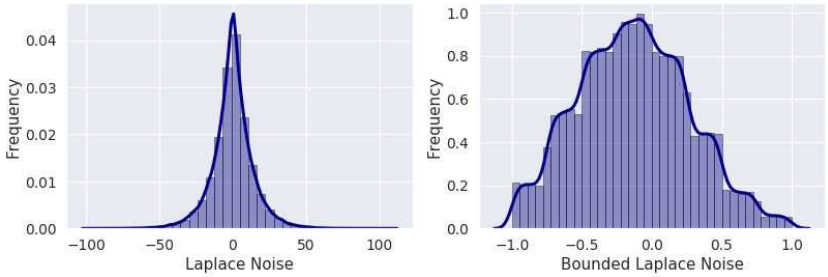
We measure the accuracy of prediction using the metric Root Mean Squared Error (RMSE) given by:

$$RMSE = \sqrt{\frac{\sum_{i=0}^{n-1} (r_i - \hat{r_i})^2}{n}} \tag{15}$$

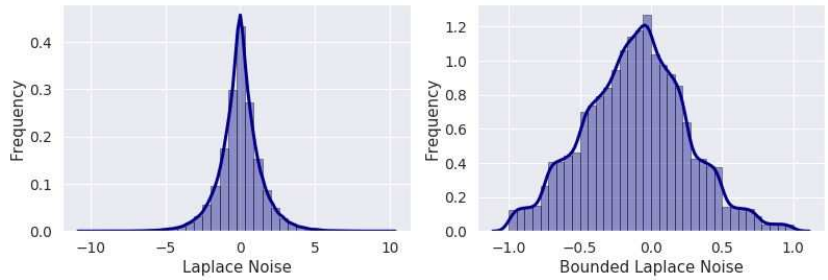
in which $r_i$ is the actual rating, $\hat{r_i}$ is the predicted rating and $n$ is the total number of ratings. We use 10-fold cross-validation to train and evaluate our proposed BLP-MoG-MF approach for both Movielens and Jester datasets over various privacy budget $\varepsilon$. The prediction accuracy is dependent on the privacy budget $\varepsilon$, higher values of $\varepsilon$ lead to weaker privacy protection levels. As there can be discrepancies while introducing noise through the Bounded Laplace mechanism, the computed RMSE is averaged across multiple runs.

### 4.3 Results

**Bounded Laplace noise distribution** In this experiment, we generated noise samples using Laplace and Bounded Laplace mechanisms. We generated 100,000 random noise samples for both mechanisms while setting their privacy budget $\epsilon$ to 0.1 and 1 respectively using Movielens dataset. Fig 2(a) and 2(b) illustrates the frequency of noise under Laplace and Bounded Laplace mechanism. The noise samples generated using Laplace mechanism approximates to a Laplace distribution while the noise samples generated using BLP produces an unknown continuous distribution. As BLP follows a conditional probability density function (see Definition.1), it no more produced noise that can approximate to a Laplace distribution. Hence, MoG is effective in estimating the noise generated by BLP.




(a) $\varepsilon = 0.1$




(b) $\varepsilon = 1$

**Fig. 2.** Laplace Vs Bounded Laplace Noise Distribution

**Prediction accuracy over various privacy budget** In this experiment, we compare the prediction accuracy of BLP-MoG-MF with other two LDP based methods. First, we compare the prediction accuracy of our BLP-MoG-MF with PG-MF [16] by varying the privacy budget $\varepsilon$ from 0.1 to 1.6 for Movielens dataset. Fig 3 shows the RMSE values for both methods and the baseline method.

As expected prediction accuracy of all the methods except the Non-private MF method improves with increase in privacy budget. Because, an increase in privacy budget implies that the magnitude of privacy leakage the mechanism allows is substantial, which in turn, leads to an increase in the utility, i.e. the prediction accuracy.

Secondly, we compare BLP-MoG-MF with ISGD [1] by varying the privacy budget $\varepsilon$ from 0.1 to 3 for both Movielens and Jester datasets. Fig 4(a) and 4(b) illustrate the RMSE values for both methods and Non-Private MF. The RMSE values of Jester dataset are larger than that of Movielens as Jester is sparser than Movielens. Fig 4(a) and 4(b) shows that the prediction accuracy of BLP-MoG-MF outperforms ISGD for all values of privacy budget $\varepsilon$. The results also show that BLP-MoG-MG produces a higher increase in prediction accuracy for Jester than Movielens for all the values of privacy budget $\varepsilon$. Jester dataset RMSE values show 35% and 28% of improvement in prediction accuracy when privacy budget $\varepsilon$ is 0.1 and 1 respectively. However, the improvement percentage for Movielens dataset is 21% and 16% for the same values of privacy budget $\varepsilon$. This implies that BLP-MoG-MF outperforms ISGD even better when the data is sparse.

**Communications Cost** We compare the communication cost of our approach to [16] and [1] in Table 1. In BLP-MoG-MF and ISGD approaches, regardless of the number of items that a user rates, the user always transmits each perturbed rating individually to the DA, once. PG-MF method requires the user to transmit only the perturbed gradient to DA. BLP-MoG-MF and ISGD methods do not require the DA to transmit any information back to the user. However, PG-MF method requires the DA to transmit updated item latent vectors matrix back to the user. The estimated transmission size for PG-MF method is approximately 0.15 MB for Movielens dataset [1], whereas BLP-MoG-MF and ISGD methods will be transmitting approximately 1 byte of data each time user send their data to DA.

**Table 1.** Communication Cost Comparison

| Method | User to DA | DA to User |
|---|---|---|
| BLP-MoG-MF | Single Perturbed Rating | No Data |
| ISGD [1] | Single Perturbed Rating | No Data |
| PG-MF [16] | Single Perturbed Gradient | Item Latent Factor Matrix |

## 5 Conclusion

In this work, we propose a local differentially private matrix factorization with mixture of Gaussian (BLP-MoG-MF) method under the consideration of an untrustworthy data aggregator. Our proposed recommendation system guarantees

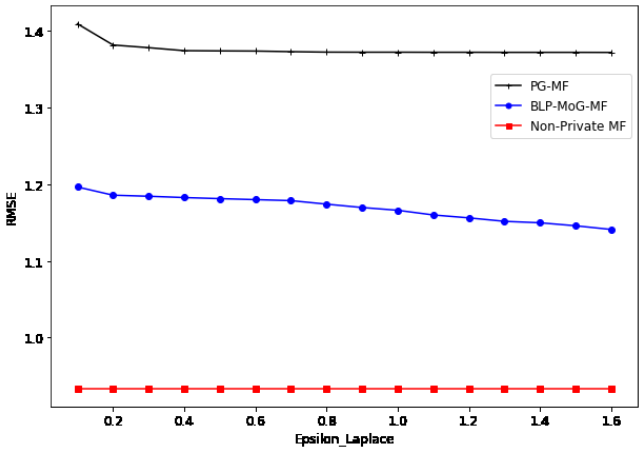
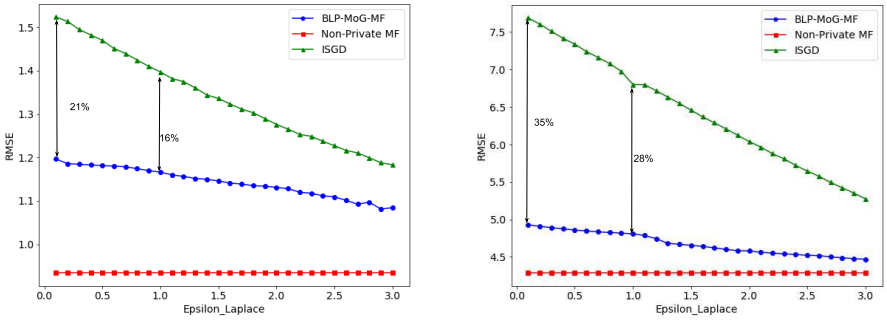



**Fig. 3.** PG-MF vs BLP-MoG-MF RMSE Comparison




(a) Movielens (b) Jester

**Fig. 4.** BLP-MoG-MF vs ISGD RMSE Comparison

strong user privacy and completely hides a user's preferences over an item from DA. It also pursues better prediction accuracy than the existing LDP based solutions [16] and [1]. Additionally, our method does not incur any additional communication cost to the user side. In future, we intend to explore approaches to improve the robustness of achieved local minima for the non-convex cost function used in MF with MoG.

## References

1. Berlioz, A., Friedman, A., Kaafar, M.A., Boreli, R., Berkovsky, S.: Applying differential privacy to matrix factorization. In: Proceedings of the 9th ACM Conference on Recommender Systems. pp. 107–114. ACM (2015)

2. Buchanan, A.M., Fitzgibbon, A.W.: Damped newton algorithms for matrix factorization with missing data. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). vol. 2, pp. 316–322. IEEE (2005)
3. De La Torre, F., Black, M.J.: A framework for robust subspace learning. International Journal of Computer Vision **54**(1-3), 117–142 (2003)
4. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the em algorithm. Journal of the Royal Statistical Society: Series B (Methodological) **39**(1), 1–22 (1977)
5. Dwork, C., Roth, A., et al.: demp. Foundations and Trends® in Theoretical Computer Science **9**(3–4), 211–407 (2014)
6. Erlingsson, Ú., Pihur, V., Korolova, A.: Rappor: Randomized aggregatable privacy-preserving ordinal response. In: Proceedings of the 2014 ACM SIGSAC conference on computer and communications security. pp. 1054–1067. ACM (2014)
7. Goldberg, K., Roeder, T., Gupta, D., Perkins, C.: Eigentaste: A constant time collaborative filtering algorithm. information retrieval **4**(2), 133–151 (2001)
8. Harper, F.M., Konstan, J.A.: The movielens datasets: History and context. Acm transactions on interactive intelligent systems (tiis) **5**(4), 19 (2016)
9. Holohan, N., Antonatos, S., Braghin, S., Mac Aonghusa, P.: The bounded laplace mechanism in differential privacy. arXiv preprint arXiv:1808.10410 (2018)
10. Hua, J., Xia, C., Zhong, S.: Differentially private matrix factorization. In: Twenty-Fourth International Joint Conference on Artificial Intelligence (2015)
11. McSherry, F., Mironov, I.: Differentially private recommender systems: Building privacy into the netflix prize contenders. In: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 627–636. ACM (2009)
12. Meng, D., De La Torre, F.: Robust matrix factorization with unknown noise. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1337–1344 (2013)
13. Narayanan, A., Shmatikov, V.: Robust de-anonymization of large datasets (how to break anonymity of the netflix prize dataset). University of Texas at Austin (2008)
14. Roy, I., Setty, S.T., Kilzer, A., Shmatikov, V., Witchel, E.: Airavat: Security and privacy for mapreduce. In: NSDI. vol. 10, pp. 297–312 (2010)
15. Shen, Y., Jin, H.: Epicrec: Towards practical differentially private framework for personalized recommendation. In: Proceedings of the 2016 ACM SIGSAC conference on computer and communications security. pp. 180–191. ACM (2016)
16. Shin, H., Kim, S., Shin, J., Xiao, X.: Privacy enhanced matrix factorization for recommendation with local differential privacy. IEEE Transactions on Knowledge and Data Engineering **30**(9), 1770–1782 (2018)
17. Srebro, N., Jaakkola, T.: Weighted low-rank approximations. In: Proceedings of the 20th International Conference on Machine Learning (ICML-03). pp. 720–727 (2003)
18. Zhu, T., Li, G., Ren, Y., Zhou, W., Xiong, P.: Differential privacy for neighborhood-based collaborative filtering. In: Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining. pp. 752–759. ACM (2013)